Ubiquitous Data Stream Mining

Mohamed Medhat Gaber[1], Shonali Krishnaswamy[1], Arkady Zaslavsky[1]
[1] School of Computer Science and Software Engineering, Monash University,
900 Dandenong Rd, Caulfield East, VIC3145, Australia
{Mohamed.Medhat.Gaber, Shonali.Krishnaswamy, Arkady.Zaslavsky}@infotech.monash.edu.au

Abstract. The dissemination of data stream systems, wireless networks and mobile devices motivates the need for an efficient data analysis tool capable of gaining insights about these continuous data streams. Ubiquitous data mining (UDM) is concerned with this problem. UDM is the time-critical process of pattern discovery in data streams in a wireless environment. In this paper, the state of the art of mining data streams is given and our approach in tackling the problem is presented. The paper also highlights the addressed and open issues in the field.

1 Introduction

Ubiquitous Data Mining (UDM) is the process of performing analysis of data on mobile, embedded and ubiquitous devices [27]. It represents the next generation of data mining systems that will support the intelligent and time-critical information needs of mobile users and will facilitate "anytime, anywhere" data mining [30], [27], [21]. The underlying focus of UDM systems is to perform computationally intensive mining techniques in mobile environments that are constrained by limited computational resources and varying network characteristics [23].

The widespread use of mobile devices with increasing computational capacity and proliferation of wireless networks is leading to the emergence of the *ubiquitous* computing paradigm that facilitates continuous access to data and information by mobile users with handheld devices. Ubiquitous computing environments are subsequently giving rise to a new class of applications termed *Ubiquitous Data Mining* (UDM), wherein the mobile user performs intelligent analysis and monitoring of data [43], [27], [17], [30]. UDM is the process of analysing data emanating from distributed and heterogeneous sources with mobile devices or within sensor networks and is seen as the "next natural step in the world of ubiquitous computing" [23]. The ever-increasing computational capacity of mobile devices presents an opportunity for intelligent data analysis in applications and scenarios where the data is continuously streamed to the device and where there are temporal constraints that necessitate analysis "*anytime, anywhere"* [30], [27]. Typical application scenarios include:

- Monitoring a stock portfolio from streamed stock market data while travelling [27].
- A travelling salesperson performing customer profiling [21].

- Continuous monitoring and analysing of status information received for intrusion detection or laboratory experiments [43].
- Analysis of data from sensors in moving vehicles to prevent fatal accidents through early detection by monitoring and analysis of status information [26]
- Performing preliminary mining of data generated in a sensor network [29]
- On-board analysis of astronomical and geophysical data [5], [37], [38]

It must be noted that ubiquitous data mining is not equivalent to performing traditional data mining tasks on a resource-constrained device, but addresses the unique needs of applications that require analysis of data in a time-critical and mobile context.
In this paper, we address the field of ubiquitous data stream mining with detailed analysis. Issues and approaches are discussed in section 2. Section 3 highlights our approach in tackling the problem of data stream mining which we have termed as *Algorithm Output Granularity* (AOG). Open issues and challenges in the field are discussed in section 4. Finally the paper is concluded in section 5.

2 Issues and Approaches

In this section, we present issues and challenges that arise in mining data streams and our approach tackles them as well as other solutions that address these challenges. Figure 1 shows the general processing model of mining data streams.

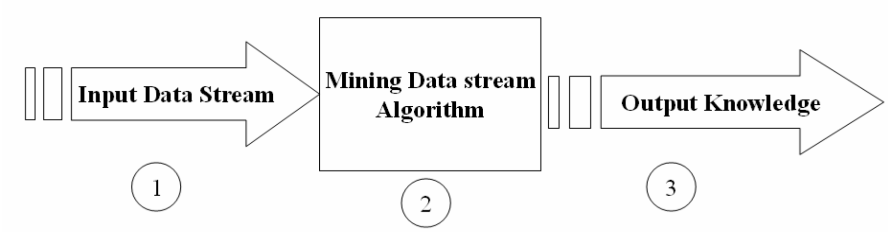

Figure 1: Mining Data Stream Process

Issues and challenges in mining data streams [2], [15], [27]:

- Handling the continuous flow of data streams.
- Minimizing energy consumption of the mobile device.
- Unbounded memory requirements due to the continuous flow of data streams.
- Required result accuracy.
- Transferring data mining results over a wireless network with a limited bandwidth.
- Data mining results' visualization on the small screen of the mobile device.
- Modeling mining results' changes over time.

- Developing algorithms for mining results' changes.
- Interactive mining environment to satisfy user requirements.

There are several strategies that address these challenges. These include [15]:

1) **Input data rate adaptation:** this approach uses sampling, filtering, aggregation, and load shedding on the incoming data elements. Sampling is the process of statistically selecting the elements of the incoming stream that would be analyzed. Filtering is the semantics sampling in which the data element is checked for its importance for example to be analyzed or not. Aggregation is the representation of number of elements in one aggregated elements using some statistical measure such as the average. While load shedding, which has been proposed in the context of querying data streams [3], [39], [40], [41] rather than mining data streams, is the process of eliminating a batch of subsequent elements from being analyzed rather than checking each element that is used in the sampling technique. Figure 2 illustrates the idea of data rate adaptation from the input side using sampling.

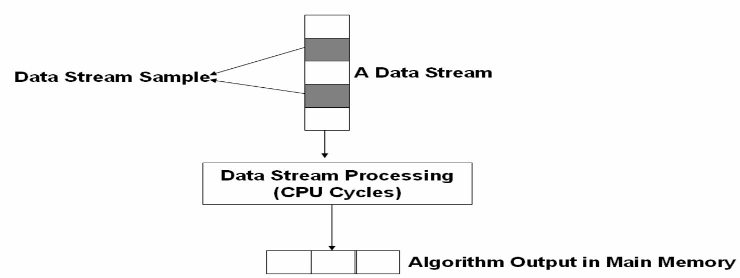

Figure 2 Data Rate Adaptation using Sampling

2) **Knowledge abstraction level:** this approach uses the higher knowledge level; that is to categorize the incoming elements into a limited number of categories and replacing each incoming element with the matching category according to a specified measure or a look-up table. This would produce fewer results conserving the limited memory. Moreover, it requires fewer number of processing CPU cycles.
3) **Approximation algorithms:** design one pass mining algorithms to approximate the mining results according to some acceptable error margin.

Mining Data Streams has been studies in [1], [4], [6], [7], [8], [9], [10], [11], [12], [13], [14], [15], [16], [17], [18], [19], [24], [25], [28], [31], [32], [33], [34], [35], [36],

[42]. Table 1. [15] summarizes the most cited data stream mining techniques according to the mining task, the used approach and the status of implementation.

Table 1 Mining Data Stream Algorithms

Algorithm	Mining Task	Approach	Status
VFKM	K-Means	Sampling and reducing the number of passes at each step of the algorithm	Implemented and tested.
VFDT	Decision Trees	Sampling and reducing the number of passes at each step of the algorithm	Implemented and tested.
Approximate Frequent Counts	Frequent itemsets	Incremental Pruning and update of itemsets with each block of transactions	Implemented and tested.
FP- Stream	Frequent itemsets	Incremental Pruning and update of itemsets with each block of transactions and time-sensitive patterns extension	Implemented and tested.
Concept-Drifting Classification	Classification	Ensemble classifiers	Implemented and tested.
AWSOM	Prediction	Incremental Wavelets	Implemented and tested (This algorithm is designed to run on a sensor). The implementation is not on a sensor.
Approximate K-median	K-Median	Sampling and reducing the number of passes at each step of the	Analytical Study

		algorithm	
GEMM	General Applied to decision tress and frequent itemsets	Sampling	Analytical study
CDM	Decision Trees, Bayesian Nets and clustering	Fourier spectrum representation of the results to save the limited bandwidth	Implemented and tested.
ClusStream	Clustering	Online summarization and offline clustering	Implemented and tested
STREAM-LOCALSEARCH	Clustering	Sampling and incremental learning	Implemented and tested against other techniques

The above approaches don't take into consideration the inherent features of data streams. The fluctuating high rate of incoming data and the resource constrained environment that the most of data stream generators characterized by. We have proposed an approach that we term algorithm output granularity in addressing this problem. AOG is an adaptive resource-aware approach that is discussed in the following section.

3 Mining Data Streams using AOG

AOG uses data rate adaptation from the output side. Figure 3 shows our strategy. We use algorithm output granularity to preserve the limited memory size according to the incoming data rate and the remaining time to mine the incoming stream without incremental integration. The algorithm threshold is a controlling parameter that is able to change the algorithm output rate according to the data rate, available memory, algorithm output rate history and remaining time for mining without integration.

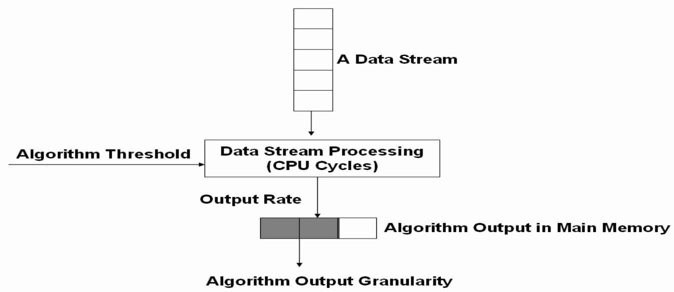

Figure 3 Algorithm Output Granularity Approach

The algorithm output granularity approach is based on the following axioms:
a) The algorithm rate (AR) is function in the data rate (DR), i.e., AR = f(DR).
b) The time needed to fill the available memory by the algorithm results (TM) is function in (AR), i.e., TM = f(AR).
c) The algorithm accuracy (AC) is function in (TM), i.e., AC = f(TM).
The controlling threshold is a parameter in each of our light-weight mining algorithm that controls the algorithm rate according to the available memory, the remaining time to fill the main memory without any incremental integration and the data rate. More details about AOG and AOG-based techniques could be found in [12], [13], [14], [15].

4- Open Issues and Challenges

There are a number of issues and challenges that have not been addressed in the previously proposed approaches. The following is a list of these issues:

- The integration between data stream management systems [20] and the ubiquitous data stream mining approaches. It is a very serious issue that should be addressed to realize a full functioning ubiquitous mining.
- The relationship between the proposed techniques and the needs of the real world applications is another important issue. Some of the proposed techniques try to get to better computational complexity with some margin error without taking care to the real needs of the applications that will use the proposed approach.
- The data pre-processing in the stream mining process should also be taken into consideration. That is how to design a very light-weight pre-processing techniques that can guarantee the quality of the mining results.
- The technological issue of mining data streams is also an important one. How to represent the data in such an environment in a compressed way? And which platforms are best to suit such special real-time applications?

- The formalization of real-time accuracy evaluation. That is to provide the user by a feedback by the current achieved accuracy with relation to the available resources.
- The data stream computing [22] formalization. The mining of data streams could be formalized within a theory of data stream computation. This formalization will facilitate the design and development of algorithms based on a concrete mathematical foundation.

5- Conclusions

The growth of data stream phenomenon and the dissemination of wireless devices motivate the need for ubiquitous data stream mining. The research in this area is in its early stages. A number of techniques and approaches have been proposed for data stream mining. This paper reviewed the state of the art and highlighted the addressed and open issues in the field. Our AOG-based mining approach has been presented briefly.

References

1. Aggarwal C., Han J., Wang J., Yu P. S.: A Framework for Clustering Evolving Data Streams. Proc. 2003 Int. Conf. on Very Large Data Bases (VLDB'03), Berlin, Germany (2003).
2. Babcock B., Babu S., Datar M., Motwani R., and Widom J.: Models and issues in data stream systems. In Proceedings of PODS (2002).
3. Babcock B., Datar M., and Motwani R.: Load Shedding Techniques for Data Stream Systems (short paper). In Proc. of the 2003 Workshop on Management and Processing of Data Streams (MPDS 2003) (2003).
4. Babcock B., Datar M., Motwani R., O'Callaghan L.: Maintaining Variance and k-Medians over Data Stream Windows. To appear in Proceedings of the 22nd Symposium on Principles of Database Systems (PODS 2003) (2003).
5. Burl M., Fowlkes C., Roden J., Stechert A., and Mukhtar S. Diamond Eye: A distributed architecture for image data mining. In SPIE DMKD, Orlando, April (1999).
6. Charikar M., O'Callaghan L., and Panigrahy R.: Better streaming algorithms for clustering problems. In Proc. of 35th ACM Symposium on Theory of Computing (STOC) (2003).
7. O'Callaghan L., Mishra N., Meyerson A., Guha S., and Motwani R.: Streaming-data algorithms for high-quality clustering. Proceedings of IEEE International Conference on Data Engineering, March (2002).
8. Cormode G., Muthukrishnan S.: What's hot and what's not: tracking most frequent items dynamically. PODS 2003. (2003) 296-306

9. Datar M., Gionis A., Indyk P., Motwani R.: Maintaining Stream Statistics over Sliding Windows (Extended Abstract). In Proceedings of 13th Annual ACM-SIAM Symposium on Discrete Algorithms (SODA 2002) (2002).
10. Domingos P. and Hulten G., A General Method for Scaling Up Machine Learning Algorithms and its Application to Clustering. Proceedings of the Eighteenth International Conference on Machine Learning, 106--113, Williamstown, MA, Morgan Kaufmann. (2001)
11. Domingos P. and Hulten G. Mining High-Speed Data Streams. In Proceedings of the Association for Computing Machinery Sixth International Conference on Knowledge Discovery and Data Mining, (2000) 71—80.
12. Gaber, M.M., Krishnaswamy, S. and Zaslavsky, A. (2004). Cost-Efficient Mining Techniques for Data Streams. In Proc. Australasian Workshop on Data Mining and Web Intelligence (DMWI2004), Dunedin, New Zealand. CRPIT, 32. Purvis, M., Ed. ACS.
13. Gaber, M, M., Krishnaswamy, S., and Zaslavsky, A., Adaptive Mining Techniques for Data Streams Using Algorithm Output Granularity, The Australasian Data Mining Workshop (AusDM 2003), Held in conjunction with the 2003 Congress on Evolutionary Computation (CEC 2003), December, Canberra, Australia, Springer Verlag, Lecture Notes in Computer Science (LNCS).
14. Gaber, M, M., Zaslavsky, A., and Krishnaswamy, S., A Cost-Efficient Model for Ubiquitous Data Stream Mining, Accepted for publication in the Tenth International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems (IPMU 2004), Perugia Italy, July 4-9.
15. Gaber, M, M., Zaslavsky, A., and Krishnaswamy, S., (2004), Towards an Adaptive Approach for Mining Data Streams in Resource Constrained Environments, Accepted for publication in the Proceedings of Sixth International Conference on Data Warehousing and Knowledge Discovery - Industry Track (DaWak 2004), Zaragoza, Spain, 30 August - 3 September, Lecture Notes in Computer Science (LNCS), Springer Verlag.
16. Ganti V., Gehrke J., Ramakrishnan R.: Mining Data Streams under Block Evolution. SIGKDD Explorations 3(2): (2002) 1-10.
17. Garofalakis M., Gehrke J., Rastogi R.: Querying and mining data streams: you only get one look a tutorial. SIGMOD Conference 2002: 635. (2002).
18. Giannella C., Han J., Pei J., Yan X., and Yu P.S.: Mining Frequent Patterns in Data Streams at Multiple Time Granularities. In Kargupta H., Joshi A., Sivakumar K., and Yesha Y. (eds.), Next Generation Data Mining, AAAI/MIT (2003).
19. Guha S., Mishra N., Motwani R., and O'Callaghan L.: Clustering data streams. In Proceedings of the Annual Symposium on Foundations of Computer Science. IEEE, November (2000).
20. Golab L. and Ozsu M. T. : Issues in Data Stream Management. In SIGMOD Record, Volume 32, Number 2, June (2003) 5-14.

21. Grossman, R., “Supporting the Data Mining Process with Next Generation Data Mining Systems”, Enterprise Systems, August 1998
22. Henzinger M., Raghavan P, and Rajagopalan S.: Computing on data streams. Technical Note 1998-011, Digital Systems Research Center, Palo Alto, CA, May (1998).
23. Hsu, J., “Data Mining Trends and Developments: The Key Data Mining Technologies and Applications for the 21st Century”, The Proceedings of the 19th Annual Conference for Information Systems Educators (ISECON 2002), ISSN: 1542-7382 (2002)
24. Hulten G., Spencer L., and Domingos P.: Mining Time-Changing Data Streams. ACM SIGKDD (2001).
25. Kargupta H.: CAREER: Ubiquitous Distributed Knowledge Discovery from Heterogeneous Data. NSF Information and Data Management (IDM) Workshop (2001).
26. H. Kargupta, R. Bhargava, K. Liu, M. Powers, P. Blair, S. Bushra, J. Dull, K. Sarkar, M. Klein, M. Vasa, and D. Handy. VEDAS: A Mobile and Distributed Data Stream Mining System for Real-Time Vehicle Monitoring. Accepted for publication in the Proceedings of the SIAM International Data Mining Conference, Orlando. (2004).
27. Kargupta, H., Park, B., Pittie, S., Liu, L., Kushraj, D. and Sarkar, K. (2002). MobiMine: Monitoring the Stock Market from a PDA. ACM SIGKDD Explorations. January 2002. Volume 3, Issue 2. Pages 37--46. ACM Press.
28. Keogh E., Lin J., and Truppel W.: Clustering of Time Series Subsequences is Meaningless: Implications for Past and Future Research. In proceedings of the 3rd IEEE International Conference on Data Mining. Melbourne, FL. November (2003) 19-22.
29. Krishnamachari B., and Iyengar S., "Bayesian Algorithms for Fault-tolerant Event Region Detection in Wireless Sensor Networks," accepted to appear in the IEEE Transactions on Computers, 2004.
30. Krishnaswamy, S., Loke, S, W., Zaslavsky, A., “Towards Anytime Anywhere Data Mining Services”, Proceedings of the Australasian Data Mining Workshop (ADM02), Held in conjunction with the 15th Australian Joint Conference on Artificial Intelligence (AI02), Canberra, Australia, 3rd December (2002).
31. Manku G. S. and Motwani R.: Approximate frequency counts over data streams. In Proceedings of the 28th International Conference on Very Large Data Bases, Hong Kong, China, August (2002).
32. Muthukrishnan S.: Data streams: algorithms and applications. Proceedings of the fourteenth annual ACM-SIAM symposium on discrete algorithms (2003).
33. Muthukrishnan S.: Seminar on Processing Massive Data Sets. Available Online: http://athos.rutgers.edu/%7Emuthu/stream-seminar.html (2003).
34. Ordonez C.: Clustering Binary Data Streams with K-means .ACM DMKD (2003).

35. Park B. and Kargupta H.. Distributed Data Mining: Algorithms, Systems, and Applications. Data Mining Handbook. Editor: Nong Ye (2002).
36. Papadimitriou S., Faloutsos C., and Brockwell A.: Adaptive, Hands-Off Stream Mining. 29th International Conference on Very Large Data Bases VLDB (2003).
37. Srivastava A. and Stroeve J.: Onboard Detection of Snow, Ice, Clouds and Other Geophysical Processes Using Kernel Methods. Proceedings of the ICML'03 workshop on Machine Learning Technologies for Autonomous Space Applications (2003).
38. Tanner S., Alshayeb M., Criswell E., Iyer M., McDowell A., McEniry M., Regner K., EVE: On-Board Process Planning and Execution, Earth Science Technology Conference, Pasadena, CA, Jun. 11 - 14, (2002).
39. Tatbul N., Cetintemel U., Zdonik S., Cherniack M. and Stonebraker M.: Load Shedding in a Data Stream Manager. Proceedings of the 29th International Conference on Very Large Data Bases (VLDB), September (2003).
40. Tatbul N., Cetintemel U., Zdonik S., Cherniack M. and Stonebraker M.: Load Shedding on Data Streams. In Proceedings of the Workshop on Management and Processing of Data Streams (MPDS 03), San Diego, CA, USA, June (2003).
41. Viglas S. D. and Naughton J.: Rate based query optimization for streaming information sources. In Proc. of SIGMOD (2002).
42. Wang H., Fan W., Yu P. and Han J.: Mining Concept-Drifting Data Streams using Ensemble Classifiers. In the 9th ACM International Conference on Knowledge Discovery and Data Mining (SIGKDD), Aug., Washington DC, USA (2003).
43. Zaki, M, J., "Editorial: Online, Interactive and Anytime Data Mining", SIGKDD Explorations, Vol. 3, Issue 2, January (2002).